Secure Communication Architecture for Dynamic Energy Management in Smart Grid

Khan, S., Khan, R., & AlBayatti, A. H. (2019). Secure Communication Architecture for Dynamic Energy Management in Smart Grid. *IEEE Power and Energy Technology Systems Journal*, *6*(1), 47-58. https://doi.org/10.1109/JPETS.2019.2891509

Published in:
IEEE Power and Energy Technology Systems Journal

Document Version:
Peer reviewed version

Secure Communication Architecture for Dynamic Energy Management in Smart Grid

Sarmadullah Khan, Rafiullah Khan, and Ali Hilal Al-Bayatti

***Abstract*—Smart grid takes advantage of communication technologies for efficient energy management and utilization. It entails sacrifice from consumers in terms of reducing load during peak hours by using dynamic energy pricing model. To enable active participation of consumers in load management, the concept Home Energy Gateway (HEG) has recently been proposed in literature. However, the HEG concept is rather new and literature still lacks to address challenges related to data representation, seamless discovery, interoperability, security and privacy. This paper presents the design of a communication framework that effectively copes with the interoperability and integration challenges between devices from different manufacturers. The proposed communication framework offers seamless auto-discovery and zero-configuration based networking between heterogeneous devices at consumer sites. It uses elliptic curve based security mechanism for protecting consumers privacy and providing best possible shield against different types of cyber-attacks. Experiments in real networking environment validated that the proposed communication framework is lightweight, secure, portable with low bandwidth requirement and flexible to be adopted for dynamic energy management in smart grid.**

***Index Terms*—Smart Grid, Dynamic Energy Management, Utility Services, Universal Plug & Play, Smart Home.**

I. Introduction

Today, efficient energy management and utilization is becoming increasingly important due to depletion of available electricity generation resources at an alarming rate [1]. Smart grid offers not only enormous opportunities for integration of renewables but also plays a vital role in distributed dynamic energy management at consumer sites [2]. Emerging smart grid services require certain sacrifice from consumers in terms of reducing load at peak demand hours in order to prevent load-shedding/blackout. This is achieved by using dynamic energy pricing model based on load forecast [3], [4], [5]. Utilities will offer higher price at peak-hours and lower price at off-peak hours [6]. To save on electricity bills, consumers will operate household appliances during off-peak hours.

Dynamic energy management requires an intelligent device at the consumer sites to enable effective command and control from the utility. The concept of Home Energy Gateway (HEG) has recently been proposed in literature for easing the development of smart grid services such as load management, home automation, integration of renewables, dynamic energy pricing, etc [7], [8]. The basic scenario is depicted in Fig. 1. The HEG communicates with the utility as well as with household appliances (e.g., heaters, TV, refrigerator, air-conditioner, washing machine, etc). It continuously tracks and controls the operational state of appliances based on user-specified configurations or real-time commands from the utility.

S.U. Khan and A.H. Al-Bayatti are with the Computer Science Department, De Montfort University, Leicester, United Kingdom e-mail: sarmadullah.khan@dmu.ac.uk, alihmohd@dmu.ac.uk

R. Khan is with the School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast, UK e-mail: rafiullah.khan@qub.ac.uk

A. Paper Motivation

This paper addresses three key challenges in the design of dynamic energy management service in smart grid: (i) seamless auto-discovery, (ii) interoperability and integration, and (iii) security and privacy.

Many consumers may not have technical skills to configure appliances and establish their communication with the HEG. Future network-enabled household appliances must operate as plug & paly without requiring any configurations from the user. Thus, the HEG and appliances need to inherent auto-discovery feature to enable seamless mutual discovery and communication as soon as connected to the network.

Household appliances are normally manufactured by different companies using custom protocols and data representations. This raises interoperability and integration issues due to lack of a standard communication framework. Thus, a common communication framework needs to be investigated and adopted by all manufacturers.

Security and privacy is always a major concern for Internet-based services. The HEG exchanges sensitive consumer information with the utility and receives commands to control household appliances. Any compromise of the HEG communication could lead to severe consequences such as increased electricity bill, unmanageable load increase on utility, blackout, operation of critical appliances at undesirable time, etc. Thus, a strong security mechanism is utmost necessary to protect communication from cyber-attacks.

B. Paper Contributions

The HEG (or gateway in general) has been developed by several researchers in literature focusing on a specific challenge e.g., auto-discovery [8], [9], interoperability [7], [9], [10], [11] or security [12], [13]. However, literature still lacks a comprehensive communication framework that could address challenges (presented in Section I-A) all together.

This paper investigates and develops a comprehensive communication framework (for the HEG and household appliances) that is equipped with all essential features presented in Section I-A (seamless auto-discovery, interoperability/integration and security). It is designed to be light-weight, highly configurable and flexible enough to be integrated in

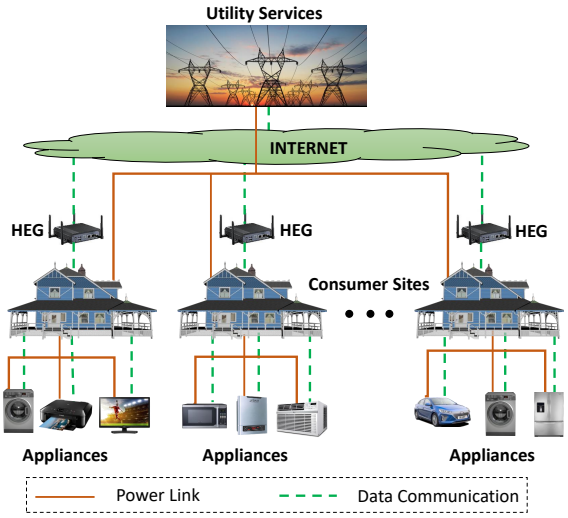


Fig. 1. Distributed dynamic energy management in smart grid.

heterogeneous appliances by different manufacturers. In short, key contributions of this paper include:

1) Design of the system architecture and clear functional and technical specifications towards practically implementing the dynamic energy management service in smart grid. This includes the investigation of basic requirements and features for the HEG, household appliances and the utility.
2) Design and development of the Universal Plug & Play (UPnP) technology for overcoming interoperability and integration issues as well as achieving two-way command and control features between the HEG and heterogeneous household appliances. This complex framework consists:
   - Design of UPnP based two-way command and control communication system architecture.
   - Implementation of Simple Service Discovery Protocol (SSDP) for enabling auto-discovery and hassle-free zero-configuration based networking between the HEG and household appliances.
   - Implementation of Simple Object Access Protocol (SOAP) for sending and receiving commands.
   - Implementation of General Event Notification Architecture (GENA) for sending and receiving notifications/information about devices operational state.
3) A suitable methodology for tracking devices operational state in a seamless manner. This includes development of a Power State Monitoring (PSM) module for seamlessly monitoring operational status of appliances from the kernel and keeping the HEG updated.
4) Design and implementation of elliptic curve based security mechanism to protect privacies of consumers and shield against different types of cyber-attacks.
5) Functional and performance evaluation of proposed system in real networking environment.

C. Paper Organization

The paper is organized as follows: Section II presents previous works from literature. Section III presents an overview of the proposed system and design challenges. Section IV presents design and characteristics of proposed communication framework. Section V describes the implementations. Analysis of the security mechanism is provided in section VI. Section VII practically evaluates the proposed system in real networking environment. Finally, Section VIII concludes the paper.

II. Related Work

Dynamic energy management is an important factor revolutionizing traditional grids into smart grids. Yao in [14] proposed a residential load management mechanism by using Photo-Voltaic (PV) cells installed on the rooftop. It schedules the operation of deferrable household appliances from peak-hours to off-peak hours and maximizes the use of local generated electricity from PV units. The excess electricity from PV units is contributed to the main grid. Gomez in [5] proposed a mechanism to forecast load and manage building heating/cooling appliances accordingly. It ensures customer satisfaction by maintaining comfortable building temperature while also helps grid to meet demand-response requirements. Several researchers have investigated load scheduling mechanisms for dynamic energy management in smart grid [15], [16], [17]. Khoury in [17] proposed a methodology to optimize energy storage under intermittent of electricity from the grid. It uses predictive scheduling to ensure continuous supply of electricity for critical household appliances.

The HEG (or gateway in general) is required at each consumer site for dynamic energy management service. The Open Service Gateway initiative (OSGi) alliance played an important role in the modular design of home gateways [18]. Several researchers have adopted the concept and developed modular gateways [8], [19], [20]. Bolla in [8] used HTTP client-server model and presented software architecture. However, presented work lacks system requirements/specifications and does not address interoperability/integration and security issues. Whereas, [19] focused on the gateway components but did not address communication aspects. Verba in [20] addressed interoperability for Internet of Things (IoT) devices but lacked plug & play, auto-discovery and security features.

Several researchers have developed gateways for energy management using MQTT protocol due to its small footprint and low bandwidth requirement [7], [11]. Lee in [7] proposed a mechanism for scheduling the operation of devices based on home energy consumption data received in the cloud. Presented work is very brief and does not include detailed technical specifications. Furthermore, proposed gateway lacks auto-discovery, zero-configuration and security features. Alternative protocol options for gateway design are XMPP [10] and CoAP [21]. XMPP provides interoperability to certain extent but lacks auto-discovery and security features. CoAP can operate over DTLS security but lacks its own built-in security mechanism.

Security threats and consequences in case of a cyber-attack have been identified by several researchers [22], [23],

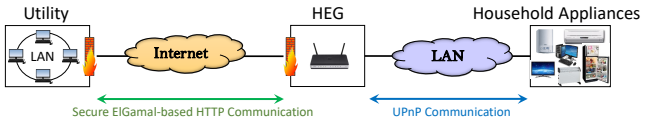


Fig. 2. Overview of proposed system.

[24]. Khan in [22] investigated different Man-In-The-Middle (MITM) attack scenarios in which a smart grid service can be compromised. Liang in [23] studied impact of data tampering or false data injection on the control algorithms. Whereas, Yao in [24] investigated stealthy cyber-attacks on smart metering infrastructure network and identified potential theft of electricity if the communication is compromised. Most protocols used in the HEG design lack security features. Thus, additional security technologies must be incorporated.

Although, different protocols and gateway designs have been studied by several researchers, literature works still lack at-least one or more of the challenges addressed in Section I-A. Thus, the objective of this paper is to investigate and develop a comprehensive communication framework that provides auto-discovery, zero-configuration based plug & play communication, interoperability, integration, privacy and security features.

III. Overview of Proposed System

The proposed system consists of utility, the HEG, household appliances as shown in Fig. 2. Communication between the HEG and household appliances is local while communication between the HEG and utility passes over public Internet. The complexity and challenges for both communication types are different. Due to private LAN, there are no security concerns for communication between the HEG and household appliances. However, proper security measures are necessary to protect privacies of consumers for communication between the HEG and utility. Therefore, communication framework is designed based on the requirements (as shown in Fig. 2) and consists of UPnP technology and ECC-based secure HTTP.

The UPnP technology offers several interesting features including hassle free zero configuration based networking between the HEG and household appliances. It enables appliances to seamlessly discover and communicate with the HEG as soon as connected to the network. The keying material and security policies for ECC-based secure HTTP communication framework are refreshed periodically in order to achieve best possible protection against cyber-attacks including cryptanalysis. These security policies consists of signature algorithm, key size, validity, authentication method, etc.

The HEG plays an important role in the design of distributed dynamic energy management service in smart grid. For energy management with better user satisfaction, the HEG can be configured to control appliances based on: (i) energy price, (ii) time frame, and (iii) operational importance of the appliance. Thinking rational economically, consumers will schedule load to off-peak hours (low energy price). Alternatively, consumers may grant control of appliances to the utility that will remotely control their operational state based on current load or energy price. The control of appliances can also be based on their operational importance e.g., heating is strictly necessary during night in winter.

Basic requirements for proposed system can be classified into three sections: (i) requirements for household appliances, (ii) requirements for the HEG, and (iii) requirements for utility.

A. Requirements for Household Appliances

Appliances need to satisfy the following basic requirements:

- They should operate as plug & play for user convenience. This includes seamless discovery and communication with the HEG without requiring any configurations.
- They must support a common set of features to achieve interoperability if manufactured by different companies.
- They should be able to operate in any network topology (e.g., bus, ring, star, mesh or hybrid).
- Each appliance should have a unique identity. Due to private IP addresses used in home networks, the Universally Unique IDentifiers (UUID) is ideal choice for identity.
- They should be able to track changes in their operational or power state and immediately notify the HEG.
- They should be able to update their operational state based on the instructions received from the HEG.
- They should support network based wake-up or switching ON features (e.g., Wake-On-LAN (WOL), Wake-on-Wireless-LAN (WoWLAN), etc).
- They should embed built-in intelligence and decide if commands from the HEG are safe enough to execute, otherwise ignore the commands.

B. Requirements for the HEG

The HEG needs to satisfy the following basic requirements:

- It should be easily accessible by utility and heterogeneous household appliances supporting common protocol.
- It should keep track of the power/operational state and capabilities of each appliance.
- It should be able to automate the use of household appliances based on the instructions received from utility.
- It should be able to send commands to appliances e.g., operation on, operation off, go to standby, etc.
- It should be able to wake-up an appliance (e.g., WOL).
- It should implement a standard Firewall to prevent unauthorized access of adversaries to the local network.
- It should implement a security mechanism for protecting communication with utility from cyber-attacks.
- It should have minimal resource requirements and comfortably operate on resource constraint gateway devices.

C. Requirements for Utility

The utility needs to satisfy the following basic requirements:

- It should keep track of each consumer site (number of appliances, power consumptions, etc). Each consumer site can be uniquely identified from the HEG's UUID.
- It should implement a security mechanism for protecting communication with consumer sites from cyber-attacks.
- It should provide current energy price and demand information to the HEG at each consumer site.
- It should be able to control appliances through the HEG (e.g., switch ON/OFF flexible appliances).

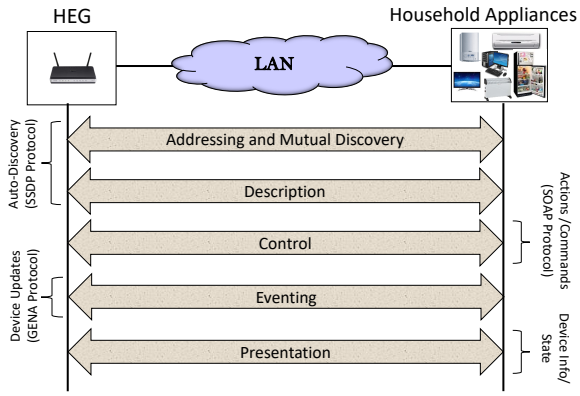


Fig. 3. UPnP semantics between the HEG and household appliances.

IV. Design of Communication Framework

This section addresses proposed UPnP and ECC-based secure HTTP communication framework (shown in Fig. 2).

A. UPnP Communication Architecture

The choice of UPnP technology for home network is motivated for several reasons: (i) auto-discovery between devices, (ii) zero-configuration based networking, (iii) it solves interoperability and integration issues between appliances from different manufacturers, (iv) it supports state variables and eventing, and (v) it is easily supported on any network device.

UPnP technology enables HEG to seamlessly discover available appliances, monitor their operation state and execute commands on them (Fig. 3 depicts basic communication semantics). As soon as an appliance is connected to the network, it acquires an IP address using DHCP protocol. The next step is discovery in which a device announces its presence in the network and also discovers the presence of other UPnP devices using SSDP protocol. After discovery, both the HEG and appliances retrieve service descriptions (capabilities, actions, commands, etc) of each other. With the knowledge of service description, the HEG and household appliances can send and receive control commands using SOAP protocol. When any state variable of an appliance changes (e.g., power state), the UPnP mechanism immediately notifies the HEG using GENA protocol. Presentation is the last step in which description of a UPnP device can be retrieved from its URL.

The UPnP device architecture [25] consists of two types of devices: Control Point (CP) and Controlled Device (CD). Functionalities of CP are similar to a client whereas, CD to a server. In general, one device implements a CP (can only send commands) and other device implements a CD (can only receive commands). In proposed system, both the HEG and household appliances implement a CP as well as a CD (to achieve two-way command & control) as depicted in Fig. 4.

To achieve interoperability, a common UPnP service needs to be implemented by household appliances (as shown in Table I). It consists of state variables (i.e., WakeMethod, PowerState, StartTime, StopTime, etc) and a list of actions that the HEG can invoke on household appliances such as: (i) *GetPowerState*: Provides current power state to the HEG,

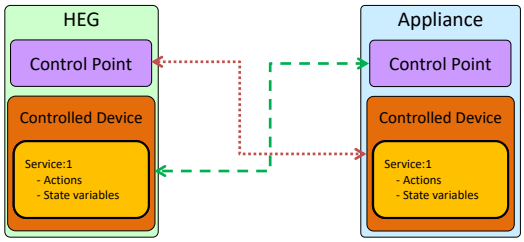


Fig. 4. Communication scenario between the HEG and household appliances.

(ii) *WakeUpMethod*: Provides information to the HEG about supported wake-up method, (iii) *OperationOn*: The appliance starts its operations, (iv) *OperationOff*: The appliance stops its operations, (v) *GoToStandby*: The appliance goes to sleep-/standby state, (vi) *StandbyPeriod*: The appliance will stay in sleep state during specified start and stop period, and (vii) *Withdraw*: Withdraws an action previously registered.

Similarly, the HEG also implements a UPnP service (as shown in Table II). It consists of actions invokable by household appliances such as: (i) *WakeUpTime*: The HEG should wake the appliance up at specified time, (ii) *NoStandbyPeriod*: The HEG should avoid putting an appliance into sleep state during the specified period, and (iii) *Withdraw*: Withdraws an action previously registered.

B. ECC-based Secure HTTP Communication

The ECC-based secure HTTP can be most suitable choice for communication between the HEG and utility due to: (i) structured meta-data, (ii) HTTP flow is normally open through Firewalls, and (iii) highest level of protection for confidentiality and integrity of communication with low computational cost. Several features of ECC based security approach make it ideal choice: (i) use of elliptic curve mechanism for establishing secret key over insecure network, (ii) use of sender private key to assure non-repudiation (iii) computationally efficient compared to Diffie Hellman [26], and (iv) suitable for real-time communication. ECC-based secure communication architecture consists of the following steps:

1) Key Generation Mechanism: The key generation procedure is shown in Algorithm 1. Before starting to generate public/private key pair, HEG/Utility first selects an elliptic curve $E_P(a, b)$. It then chooses a point on this elliptic curve i.e., E_1 and a random number R. This random number is an additive factor that shows how many times E_1 must be added with itself to generate E_2. HEG/Utility keeps this random number secret as its private key and announces E_1, E_2 and P as public key. Due to the discrete logarithmic problem and modulus operation, it is not possible to guess/calculate the value of R from E_1 and E_2. This is because, different values of R result in same E_2 for the value of E_1. For example, an elliptic curve with $P = 13$, $E_1 = 7$ and $R = 3$ results $E2 = RE1 \ mod \ P = 8$. If the value of R changes to $R = 16$, it results again $E2 = RE1 \ mod \ P = 8$.

2) Secure Communication: For secure communication, a message is first encrypted and then its hash (i.e., MAC) is calculated. This approach helps the receiver to verify the

TABLE I
ACTIONS PROVIDED BY UPNP POWER MANAGEMENT SERVICE OFFERED BY HOUSEHOLD APPLIANCES. IN MEANS 'PARAMETERS/VALUES SENT TO THE RECEIVING DEVICE' WHILE OUT MEANS 'PARAMETERS/VALUES RETURNED BACK TO THE SENDER'.

	Arguments	UUID	Address	PowerState	WakeMethod	StartTime	StopTime	RegID
	Allowed Values	–	IPv4/IPv6	ON/OFF/Standby	WOL/WoWLAN	HH:MM:SS	HH:MM:SS	Integer
Action Name	GetPowerState	IN	IN	OUT	-	-	-	OUT
	WakeUpMethod	IN	IN	-	OUT	-	-	OUT
	OperationOn	IN	IN	-	-	-	-	OUT
	OperationOff	IN	IN	-	-	-	-	OUT
	GoToStandby	IN	IN	-	-	-	-	OUT
	StandbyPeriod	IN	IN	-	-	IN	IN	OUT
	Withdraw	IN	IN	-	-	-	-	IN

TABLE II
ACTIONS PROVIDED BY UPNP SERVICE OFFERED BY THE HEG. IN MEANS 'PARAMETERS/VALUES SENT TO THE RECEIVING DEVICE' WHILE OUT MEANS 'PARAMETERS/VALUES RETURNED BACK TO THE SENDER'.

	Arguments	UUID	Address	Time	StartTime	StopTime	RegID
	Allowed Values	–	IPv4/IPv6	HH:MM:SS	HH:MM:SS	HH:MM:SS	Integer
Action Name	WakeUpTime	IN	IN	IN	-	-	OUT
	NoStandbyPeriod	IN	IN	-	IN	IN	OUT
	Withdraw	IN	IN	-	-	-	IN

Algorithm 1 Key Generation

1: **procedure**
2: *Select an elliptic curve* $E_P(a, b)$
3: *Select a point* E_1 *on* $E_P(a, b)$
4: *Select a random number R*
5: *Calculate* $E_2 = RE_1 \mod P$
6: *Keep R secret as private key*
7: *Make (* E_1 *,* E_2*,* $E_P(a, b)$*) public*
8: **end procedure**

message first and then decrypt to save computational resources (a fake/corrupted message is simply discarded without decryption). The proposed one-to-one secure communication model works as follow:

- For the HEG to communicate with the $Utility$, it requests the public key of $Utility$ from the central Key Management Center (KMC).
- HEG encrypts the message (M) using its own randomly generated number (R_{HEG}) and public key parameters of $Utility$ (E_{U1}, E_{U2}, E_P) where $E_{U2} = R_{Utility}E_{U1}$ as follow:

$$C_1 = R_{HEG}E_{U1} \ mod \ P \tag{1}$$

$$C_2 = M + R_{HEG}E_{U2} \ mod \ P \tag{2}$$

where C_1 and C_2 are the cipher text generated from the message M using the utility public key parameters E_1, E_2 and a random number R_{REG}.
- Once the message is encrypted, HEG generates a MAC from the encrypted message using a hashing function H and signs it with its own private key R_{HEG} as

$$MAC = \{H(C_1 + C_2)\}_{R_{HEG}} \tag{3}$$

- HEG sends C_1, C_2 and MAC to $Utility$

$$HEG \rightarrow Utility : [C_1 \ || \ C_2 \ || \ MAC] \tag{4}$$

- $Utility$ creates MAC' from the received message. It also decrypts the received MAC using the HEG public key K_{HEG} and compares it with the created MAC' as

$$MAC' = H(C_1 + C_2) \tag{5}$$

$$MAC = \{H(C_1 + C_2)_{R_{HEG}}\}_{K_{HEG}} = H(C_1 + C_2) \tag{6}$$

- If received decrypted MAC is equal to new calculated MAC', it accepts and decrypts the entire message as

$$M = C_2 - (R_{Utility} \ \times \ C_1) mod P \tag{7}$$

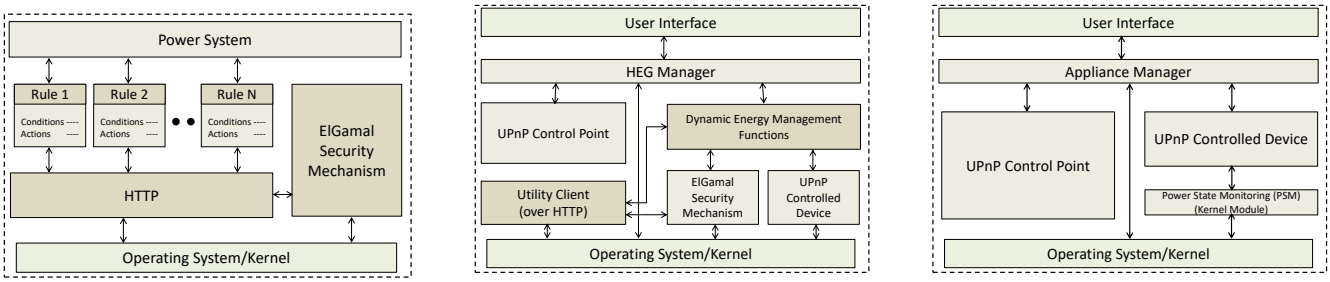
$$M = M + R_{HEG}R_{Utility}E_{U1} - R_{Utility}R_{HEG}E_{U1} \ mod \ P \tag{8}$$

$$M = M \tag{9}$$

otherwise it discards the message.

V. IMPLEMENTATIONS

The generic architecture of software entities for utility, HEG and household appliances is shown in Fig. 5. The Power System block in utility software (see Fig. 5(a)) is presented in generic way and its functionalities are based on the smart grid service. For dynamic energy management service under consideration, Power System block provides current energy price and demand information. The behavioral rules are invoking different actions based on the information received from Power System block (e.g., informs the HEG to reduce load to certain threshold). The HTTP block implements client and server for two-way communication with the HEG. Communication security is achieved using ElGamal ECC.



(a) Utility (b) HEG (c) Appliance

Fig. 5. Basic software structures in the proposed communication framework.

The functional blocks of the HEG software are shown in Fig. 5(b). It embeds a UPnP CP as well as a CD for two way communication with household appliances. The CP is used for sending commands whereas CD receives commands from household appliances. The Utility Client is used for communication with utility and provides it access to the local implementation of energy management service.

The functional blocks of the software for household appliances are shown in Fig. 5(c). Similarly to the HEG, it also embeds a UPnP CP as well as a CD for two way communication with the HEG. It implements all of the actions specified in Section IV. The home appliance also implements a PSM kernel module. The PSM module continuously monitors all changes in the operation or power state (e.g., switch ON, switch OFF, sleep state, etc) of the device and immediately notifies the HEG over UPnP communication framework.

All three software entities in Fig. 5 were implemented using C/C++ programming language in the Linux operating system. The libupnp libraries were used for the implementation of UPnP communication framework. For ease in implementation of network tasks, boost libraries were used.

VI. Security Analysis

To validate the security features, first we describe the adversary/threat model to highlight the attacker's capabilities. Then the proposed security mechanism is analyzed against various attacks based on the attacker capabilities.

A. Adversary Model

We consider an adversary with the following capabilities: (i) it has full access to the network communications, and (ii) it can listen, capture, store, modify, replay, delay and drop messages (packets). The different types of attacks that we consider in this article are the threats to information/key messages exchanged between two devices.

B. Security Attacks

Messages must be secured from the understanding, modification and replication by an attacker. Such types of attacks are called communication-based attacks and they include:

1) Sybil Attacks: In this type of attack, attacker creates multiple fake nodes that carry the authentic node IDs. But these fake nodes have only the public keys of authentic nodes and have no information regarding the private keys. Once these fake nodes receive the messages that are encrypted by public keys, they cannot decrypt it. This is because, they need private keys for decrption. But private keys are only known to the authentic devices and only the authentic devices are able to respond to the message. In the proposed framework, HEG uses the public private key approach, hence an attacker cannot succeed by creating fake HEGs as these fake HEGs will not be able to decrypt and respond to correct queries.

2) Man in the Middle Attack: As the name implies, in these attacks the attacker manages to intercept all exchanges between two communicating parties without revealing its real identity to either of them. Hence, in order to succeed, the attacker needs to successfully impersonate each communicating party in the session. However, in our framework each device is authenticated based on their public private key pair which prevents its impersonation. For example, if an attacker wants to impersonate an authentic device during the communication between KMC and HEG, the attacker cannot sign a hash generated for each message using the authentic HEG/KMC private key. This is because the private key is only known to the authentic HEGs/KMC.

3) Authentication Attacks: During the authentication phase, sender uses its own public/private key pair and receiver public key to verify and authenticate each other. To do so, sender creates the hash of a message and sings it by its own private key. Sender encrypts the message along with its signed hash using receiver public key and send it to the receiver. Receiver upon receiving the message, it decrypts the message using its own private key to extract the message and signed hash. Receiver decrypts the signed hash using the sender public key. Receiver also generates another hash from the received message and compare it with the received signed hash. If both are equal, message is accepted otherwise rejected. In the propose framework, HEGs use their private keys to sign and encrypt the hash and public keys of the receiver to encrypt the message (i.e. C_1 and C_2). If attacker tries to authenticate it self to the HEG, HEG is not the able to decrypt the messages because HEG is not able to get the public key of attacker from the KMC which registers all HEGs public keys with itself. This fails the authentication between HEG and an attacker

TABLE III
SECURITY ANALYSIS OF HTTP-BASED COMMUNICATION BETWEEN UTILITY AND THE HEG.

		Without security	With security
CIA Model	Confidentiality	None	Strong
	Integrity	None	Strong
	Availability	Vulnerable	Vulnerable
Attack Type	Reconnaissance	Vulnerable	Protected
	Authentication/Access	Vulnerable	Protected
	Man In The Middle (MITM)	Vulnerable	Protected
	Replay / Reflection	Vulnerable	Protected
	Denial of Service (DoS)	Vulnerable	Vulnerable

and protects the system from outsiders.

4) Replay attacks: These are implemented by resending at a later time some messages recorded from a previous legitimate message exchange, in order to gain access to protected resources or to privileges. Proposed framework prevents this type of attacks by using timestamping each message and then checking for its freshness. If a message is received within the time frame specified in packet, it is accepted otherwise rejected as this could be a replayed or delayed message.

Proposed elliptic curve based secure HTTP communication uses encryption to achieve protection against reconnaissance. Authentication attacks can also be prevented as the adversary cannot get access to security credentials. Without knowledge of keying material and security policies, eavesdropping on network traffic cannot be possible. Absolute protection against DoS attack is impossible for any security mechanism, however, proposed approach can significantly reduce its impact by using MAC verification step first. The cookie helps recipient easily detect DoS and discards packet without processing (saves memory and CPU resources). The effectiveness of proposed system is summarized in Table III.

VII. EVALUATION AND PERFORMANCE ANALYSIS

The testbed for experimental evaluation consists of a utility communicating with different consumer sites as depicted in Fig. 6. The HEG software is executed on a low power pocket PC i.e., Raspberry Pi v2 (ARMv6 700 MHz, 512 MB). This prevents any incremental energy waste as Raspberry Pi has full load power consumption of just 3.8 W. A standard PC is considered as a typical home appliance due to unavailability of networking features on legacy household appliances. However, it is expected that heaters, TV, lightening, refrigerators, air conditioners and other household appliance will sooner or later become part of the network.

The first step in experimental evaluation is the functional verification of all three software entities in Fig. 5. The UPnP communication architecture was efficient and devices seamlessly discovered each other as soon as connected to the network. All UPnP actions reported in Table I and Table II were successfully verified. The PSM kernel module was also reliably tracking changes in the appliance operational status. The HEG was able to alter remotely the operational state of household appliances whenever necessary (i.e., triggering built-in OS calls). The WOL feature that enables the HEG to remotely wake-up an appliance was also correctly verified.

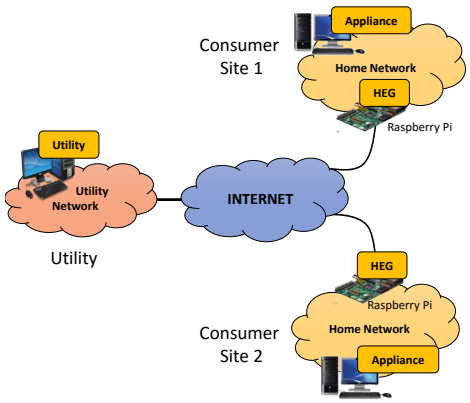


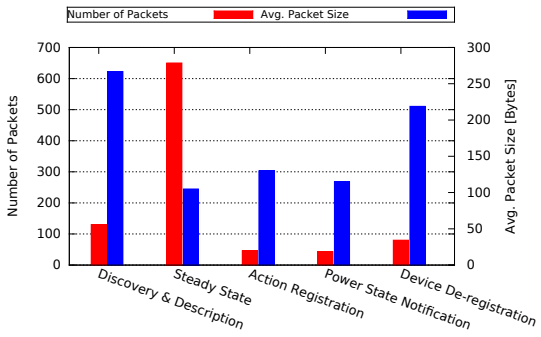
Fig. 6. Experimental Testbed.

The next step in evaluation is the analysis of different performance factors which may can impair the real-time operations. The following subsections analyzes critical factors such as latencies, resource requirements, overhead, etc.

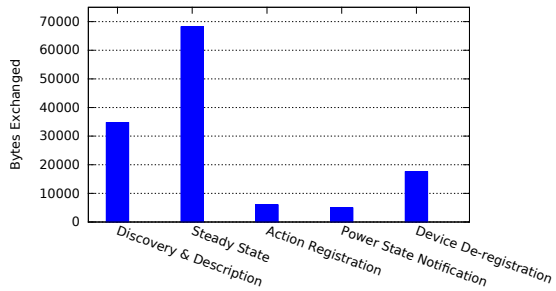
A. Communication Overhead

Communication overhead is a critical factor that can affect throughput and real-time operations for any communication framework. It is also a factor determining the required link bandwidth. High overhead leads to slower throughput on low bandwidth links. For overhead analysis, we considered each individual UPnP event such as discovery, invoking actions, notification of state changes and de-registration. The total experiment duration was 12 minutes during which a sequence of activities take place (i.e., appliance registers with the HEG, HEG remotely alters power state of the appliance, HEG remotely wakes up the appliance, HEG de-registers with the appliance). The overhead in terms of total number of packets exchanged is shown in Fig. 7(a). The majority of packets were exchanged during steady state which are periodic presence advertisements with smallest average packet size. The average packet size during discovery and de-registration is big due to carrying complete UPnP device and service descriptions. The average packet size during power state notification is very small. Further, notification messages are very infrequent. For more clarity, Fig. 7(b) depicts the total number of Bytes exchanged during different events which is linked directly with the average packet size and packets transmission rate.

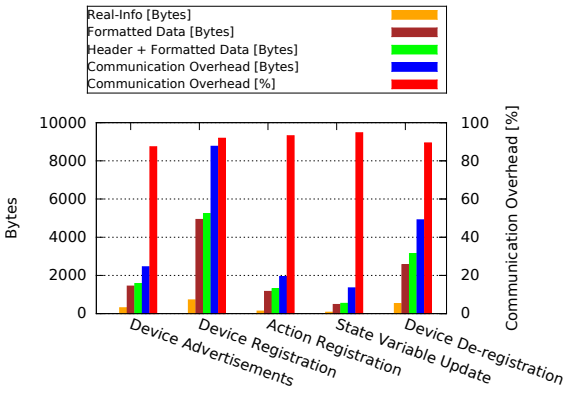
Fig. 7(c) provides more detailed overhead information by classifying packet content into real information, overhead Bytes due to UPnP formatting, overhead Bytes due to headers and total communication semantics overhead. It is obvious that real information in packets is high during registration and de-registration phases due to downloading of UPnP device and service descriptions. The percentage of real information is very low during action registration and state variable update. Thus, it can be concluded that small size packet are frequent and large size packets are rarely exchanged based on event types. Even though, the percentage of overhead Bytes in each packet is high, it does not affect reliability due to small size packets.



(a) Number of packets exchanged and average packet size during different UPnP events



(b) Total Bytes exchanged during different UPnP events



(c) Percentage overhead during different UPnP events

Fig. 7. UPnP overhead analysis in proposed framework.

B. Resource Requirements

The resource requirements were analyzed in terms of CPU, memory and bandwidth. The CPU usage was always less than 10 % even on low power PC, the Raspberry Pi. This is due to the fact that no much CPU intensive tasks are involved in the proposed system. The memory requirement can be more critical for HEG as the legacy gateway devices are equipped with 16 MB or 32 MB. However, the observed memory requirement was very low (as reported in Table IV) which makes HEG easily portable to any legacy gateway device. Comparing to utility and home appliances, the HEG memory requirement is slightly higher due to the implementation of both, ECC based security mechanism and UPnP communication framework. The memory requirement for utility software

TABLE IV
MEMORY REQUIREMENT OF DEVELOPED SOFTWARE ENTITIES.

Utility Software	HEG Software	Appliance Software
2.98 MB	4.87 MB	4.53 MB

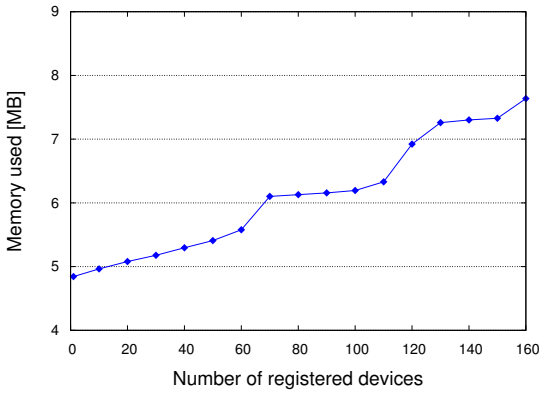


Fig. 8. The HEG memory requirement vs number of registered appliances.

is very low due to implementation of basic set of features. However, implementation of more complex utility software could slightly increase the memory requirements.

Note that Table IV presents memory requirement for the HEG with a single registered home appliance. The resource requirement for the HEG will increase with the increase in the number of registered appliances. In a realistic scenario, the HEG will be simultaneously managing several home appliances. Fig. 8 analyzes the memory requirement for the HEG with increase in the number of registered home appliances. It can be observed that the memory requirement is lower than 8 MB even with 160 registered appliances (much higher number than in a realistic home scenario). Table V presents memory requirement for the HEG software for each additional home appliance. Each appliance on average requires 17.86 KB of additional memory at the HEG. Due to very low memory requirement, the HEG software on a legacy gateway device can easily manage hundreds of home appliances.

For the HEG to manage large number of appliances, the volume of network traffic will also increase. This leads to additional computation and increases the CPU requirement for the HEG. It is a critical factor that could potentially limit the HEG scalability for managing large number of appliances (depending on the specific hardware platform). To analyze the robustness and scalability of the HEG, a high volume of network traffic has been generated at the HEG. Fig. 9 depicts the CPU usage under varying network traffic load. It can be observed that even on a low power PC (i.e., Raspberry Pi), the CPU usage is always less than 30 % even at high volume of network traffic. The concludes that the HEG is scalable and suitable to be deployed on legacy gateway devices.

The bandwidth requirement is very critical for any network protocol. Low available link bandwidth than requirement can cause traffic congestion and packet loss. The bandwidth requirement for developed communication framework is reported in Table VI for a single registered device/appliance.

TABLE V
ADDITIONAL REQUIRED MEMORY PER DEVICE/APPLIANCE FOR THE HEG. THE RESULTS ARE AVERAGED OVER 100 TRIALS.

Min (KB)	Avg (KB)	Max (KB)	Mean Dev (KB)
14.62	17.86	20.14	0.837

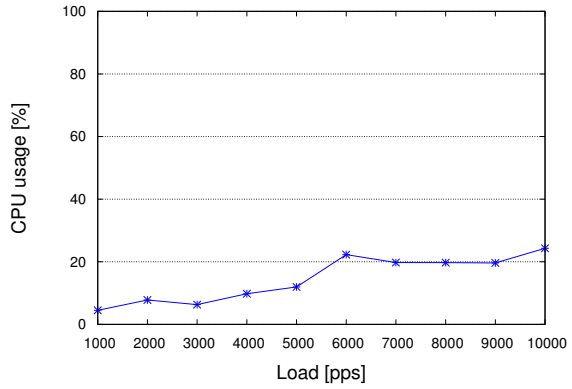


Fig. 9. The HEG CPU usage under varying network traffic load.

Bandwidth requirement also depends on the number of registered appliances and ideally will scale linearly. Fig. 10 depicts the minimum required bandwidth versus increasing number of registered appliances. Due to higher overhead, bandwidth requirement for UPnP communication framework is slightly higher compared to HTTP protocol. However, the requirement is significantly lower than the supported data rate by most of the access technologies (e.g., ADSL Lite: 1.5 Mbps, wireless 802.11b: 11 Mbps, etc). Even for 100 registered appliances, bandwidth requirement is less than 150 kbps. Due to very low resource requirements, smooth and reliable operations can be easily achieved on today's access technologies.

C. Communication Latencies

Latencies can leave adverse impact on the operations of real-time applications. In experiments, latencies were classified into two types: communication latencies and processing latencies. Low communication latencies are very important for reliable operations and depends on the available bandwidth. It is necessary that power state notifications should be transfered with shortest possible delay due to gap of only 1-2 seconds between kernel notification and actual change in device power state. Further, updates on security polices should be received by the HEG with lowest possible delay before the expiry of old credentials. Table VII reports observed communication latencies for developed communication framework. It can be observed that UPnP communication latency is very low and does not leave any adverse impact on the operations. However, the reported HTTP communication latency is slightly high due to use of external third party DNS service (i.e., NO-IP DNS) in experiments. The latency can be significantly reduced by designing a private DNS service.

TABLE VI
BANDWIDTH REQUIREMENT FOR DEVELOPED COMMUNICATION FRAMEWORK.

UPnP Communication	HTTP Communication
1.48 kbps	0.41 kbps

TABLE VII
COMMUNICATION LATENCIES ANALYSIS.

UPnP Communication	HTTP Communication
96.33 ms	1.26 sec

Processing latencies of developed software entities depend significantly on the ECC-based encryption and signature algorithm. Fig. 11 depicts the elliptic curve based encryption latencies with increasing message size. The latencies are significantly low even for large messages. Similarly, measured latencies for signature calculation are also very low as depicted in Fig. 12. Thus, processing latencies of software entities are always less than 20 ms and does not leave any negative impact on the operations.

The computational requirement for the HEG will increase with the increase in the number of registered appliances which might affect the processing latencies. To analyze a realistic scenario, communication and processing latencies for the HEG have been observed in Fig. 13 for increasing number of registered appliances. For clarity, latencies have been measured for different stages of UPnP communication

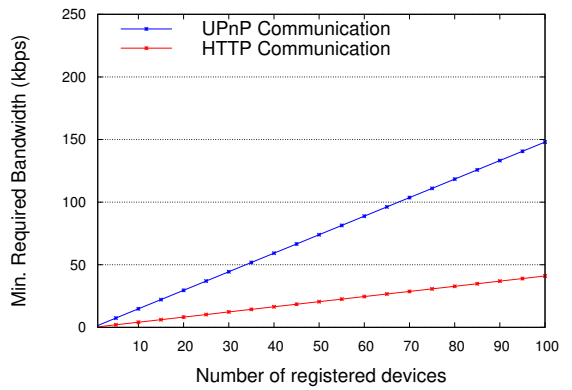


Fig. 10. Bandwidth requirement vs number of registered appliances.

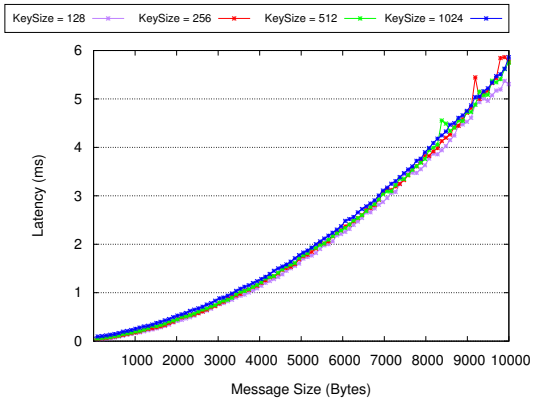


Fig. 11. ECC based encryption latencies with increasing message size.

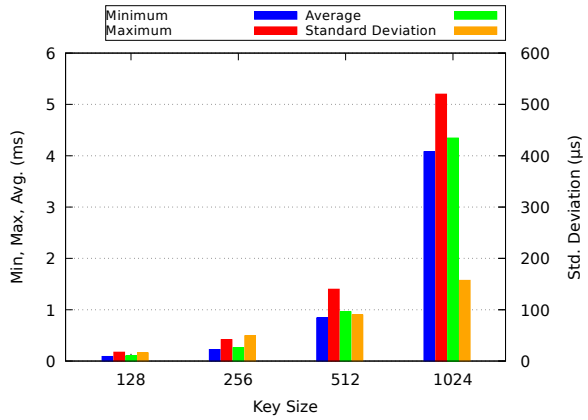


Fig. 12. Signature latencies averaged over 100 trials.

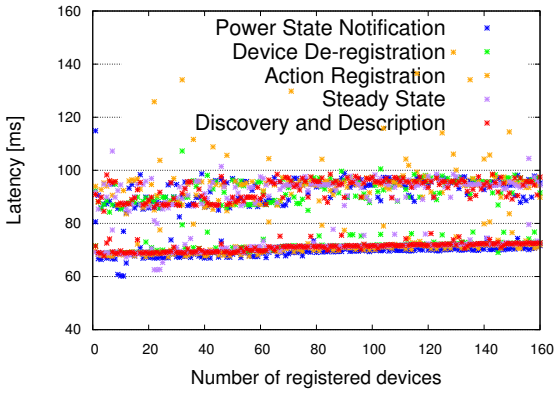


Fig. 13. Observed latencies vs number of registered appliances. The lower band of values come from HEG processing, whereas the higher band also include UPnP communication latency.

semantics. It can be observed in Fig. 13 that increase in the number of household appliances has negligible impact on the observed latencies. Even for 160 appliances (much higher number than a realistic home scenario), UPnP latencies are less than 100 ms on a low power Raspberry Pi. Fig. 14 depicts the average processing latency of the HEG for commands received from the utility. Note, Fig. 14 does not consider the HTTP communication latency as it depends on the access technology used by the customers. It can be observed in Fig. 14 that the HEG processing latency is less than 20 ms for commands received from the utility even with 160 registered household appliances. Due to low processing requirement, HEG is scalable and can easily offer service for hundreds of household appliances even on low power gateway devices.

VIII. Conclusions

Dynamic energy management has several benefits for consumers as well as power grids. Cutting off un-necessary loads or scheduling them from peak demand hours to off-peak hours not only eases pressure on power grids but also offers economic benefits to consumers. Several researchers have proposed an intelligent device at consumer sites (i.e., the HEG) and addressed specific challenges such as auto-discovery [8], [9], interoperability [7], [9], [10], [11] or security [12], [13]. However, published works lack a comprehensive communication framework design that could address these challenges all together.

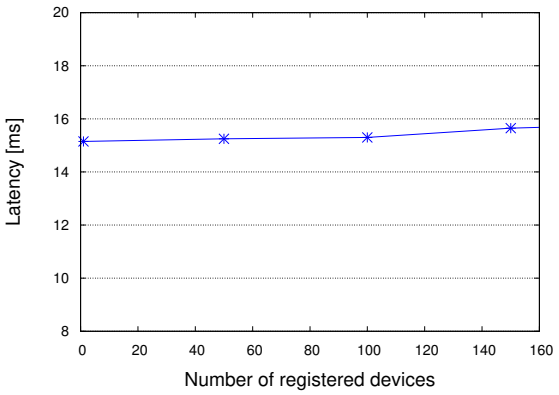


Fig. 14. Average HEG processing latency for commands received from the utility with increasing number of registered appliances.

This paper proposed a comprehensive communication framework based on the UPnP technology that provides auto-discovery, zero-configuration based plug & play networking, interoperability, integration, privacy and security features. In particular, basic requirements and key challenges have been identified for the utility, HEG and household appliances in the design of distributed dynamic energy management service in smart grid. This paper proposed a communication framework that uses UPnP technology for local communication between the HEG and household appliances whereas, ECC based secure HTTP protocol for communication between the HEG and utility. The use of UPnP technology provides numerous benefits including hassle free seamless networking with zero-configuration, auto-discovery, interoperability and integration between the HEG and heterogeneous household appliances. Experiments in real networking environment verified the functionalities, suitability and effectiveness of the developed UPnP communication architecture for dynamic energy management in smart grid.

This paper also presented an effective security mechanism based on elliptic curve approach for protecting communication between the HEG and utility against different types of cyber-attacks. It is experimentally validated that the proposed communication framework is lightweight, secure, portable with low bandwidth requirement and flexible enough to be adopted for any emerging smart grid service involving consumers. This will enable rapid adoption of new smart grid control and monitoring applications with increased consumers participation, reduced risk of blackout/load-shedding and reliable low-cost supply of electricity.

References

[1] M. H. Yaghmaee, M. S. Kouhi, A. L. Garcia, Personalized Pricing: A New Approach for Dynamic Pricing in the Smart Grid, in: IEEE Smart Energy Grid Engineering (SEGE), 2016.

[2] M. Shakeri, et al., An Intelligent System Architecture in Home Energy Management Systems (HEMS) for Efficient Demand Response in Smart Grid, in: Elsevier Energy and Buildings, Vol. 138, pp: 154-164, 2017.

[3] Q. Tang, K. Yang, D. Zhou, Y. Luo, F. Yu, A Real-Time Dynamic Pricing Algorithm for Smart Grid With Unstable Energy Providers and Malicious Users, in: IEEE Internet of Things, Vol. 3, issue: 4, 2016.

[4] A. R. Khan, et al., Load Forecasting, Dynamic Pricing and DSM in Smart Grid: A Review, in: Elsevier Renewable and Sustainable Energy Reviews, Vol. 54, pp: 1311-1322, 2016.

[5] J. A. Gomez, M. F. Anjos, Power Capacity Profile Estimation for Building Heating and Cooling in Demand-Side Management, in: Elsevier Journal on Applied Energy, Vol. 191, pp: 492-501, 2017.

[6] M. L. Tuballa, M. L. Abundo, A Review of the Development of Smart Grid Technologies, in: Elsevier Journal on Renewable and Sustainable Energy Reviews, Vol. 59, pp: 710-725, 2016.

[7] C. H. Lee, Y. H. Lai, Design and Implementation of a Universal Smart Energy Management Gateway based on the Internet of Things Platform, in: IEEE Int. Conf. on Consumer Electronics (ICCE), 2016.

[8] R. Bolla, M. Giribaldi, R. Khan, M. Repetto, Design of home energy gateway boosting the development of smart grid applications at home, in: Conf. on Energy Aware Computing Systems and Applications, 2013.

[9] R. Klauck, Seamless Integration of Smart Objects into the Internet Using XMPP and mDNS/DNS-SD, in: PhD Dissertation, Brandenburg University of Technology Cottbus-Senftenberg, Germany, 2016.

[10] S. K. Viswanath, et al., System Design of the Internet of Things for Residential Smart Grid, in: IEEE Wireless Communications Journal, Vol. 23, issue: 5, pp: 90-98, 2016.

[11] Y. Upadhyay, A. Borole, D. Dileepan, MQTT Based Secured Home Automation System, in: Colossal Data Analysis and Networking, 2016.

[12] P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttila, M. Sain, Lightweight and Secure Session-Key Establishment Scheme in Smart Home Environments, in: IEEE Sensors Journal, Vol. 16, issue: 1, 2016.

[13] Y. Gong, Y. Cai, Y. Guo, Y. Fang, A Privacy-Preserving Scheme for Incentive-Based Demand Response in the Smart Grid, in: IEEE Transactions on Smart Grid, Vol. 7, issue 3, pp: 1304-1313, 2016.

[14] E. Yao, P. Samadi, V. W. S. Wong, R. Schober, Residential Demand Side Management Under High Penetration of Rooftop Photovoltaic Units, in: IEEE Transactions on Smart Grid, Vol. 7, issue:3, pp: 1597-1608, 2016.

[15] B. Zhou, et al., Smart Home Energy Management Systems: Concept, Configurations, and Scheduling Strategies, in: Elsevier Joural on Renewable and Sustainable Energy Reviews, Vol. 61, pp: 30-40, 2016.

[16] S. J. Lee, et al., Supply and Demand Management System based on Consumption Pattern Analysis and Tariff for Cost Minimization, in: Int. Conf. on Advanced Communication Technology, 2016.

[17] J. Khoury, R. Mbayed, G. Salloum, E. Monmasson, Predictive Demand Side Management of a Residential House under Intermittent Primary Energy Source Conditions, in: Energy and Buildings, Vol. 112, 2016.

[18] OSGi Alliance. Available at: https://www.osgi.org.

[19] F. Ding, et al., A Smart Gateway Architecture for Improving Efficiency of Home Network Applications, in: Journal of Sensors, 2016.

[20] N. Verba, et al., Platform as a service gateway for the Fog of Things, in: Elsevier Journal on Advanced Engineering Informatics, 2016.

[21] C. Pham, Y. Lim, Y. Tan, Management Architecture for Heterogeneous IoT Devices in Home Network, in: IEEE Consumer Electronics, 2016.

[22] R. Khan, P. Maynard, K. McLaughlin, D. Laverty, S. Sezer, Threat Analysis of BlackEnergy Malware for Synchrophasor Based Real-time Control and Monitoring in Smart Grid, in: ICS-CSR, 2016.

[23] G. Liang, J. Zhao, F. Luo, S. Weller, Z. Y. Dong, A Review of False Data Injection Attacks Against Modern Power Systems, in: IEEE Transactions on Smart Grid, Vol. PP, issue: 99, 2017.

[24] J. Yao, et al., Network Topology Risk Assessment of Stealthy Cyber Attacks on Advanced Metering Infrastructure Networks, in: 51st Annual Conference on Information Sciences and Systems (CISS), 2017.

[25] Open Connectivity Foundation. Available at: https://openconnectivity.org.

[26] W. Diffie, M. Hellman, New directions in cryptography, IEEE Transactions on Information Theory 22 (6) (1976) 644–654.